first SQL database call is different than a second SQL database call generated to a SQL database in response to executing the general computer language programming call.

23.   (Amended) The method of claim 19 wherein said database protocol command is an SQL call.

26.   (Amended) A computer readable medium, for accessing a database, comprising instructions which when executed by a computer, comprises the following steps:

exposing software components, in a first computer programming language, of an application server as database elements, said software components being operative for accessing said database;

receiving database a protocol command for accessing the database at the application server;

converting the database protocol command to a command syntax of the first computer programming language corresponding to at least a selected one of said software components; and

accessing said database using said selected one of said software components.

REMARKS

In the Office Action dated September 24, 2002, the Examiner rejected claims 1, 3-16, 18-23, 26, 27, 29-37, and 40 under 35 U.S. C. § 103(a) as being unpatentable over Hembry (U.S. Patent no. 5,890,160) and rejected claims 2, 17, 24, 25, 28, 38, and 39 under 35 U.S.C. § 103(a) as being unpatentable over Hembry in view of Moore et al.

(U.S. Patent No. 6,338,068). Based on the following remarks, Applicant respectfully traverses the rejections of claims 1-40 under 35 U.S.C. § 103(a).

By this amendment, Applicant has amended claims 16, 18, 23, and 26 to correct minor deficiencies associated with these claims. Since the Examiner examined these claims as if they were drafted in accordance with the amendment herein, Applicant submits that the amendments to claims 16, 18, 23, and 26 have not been made for reasons of patentability.

Applicant respectfully traverses the rejection of claims 1, 3-16, 18-23; 26, 27, 29-37, and 40 under 35 U.S. C. § 103(a) because Hembry fails to teach or suggest every recitation of these claims.

Hembry teaches a system for facilitating database access operations through the use of object oriented interface objects. In particular, Hembry discloses a database access process that uses an Object Oriented (OO) interface between an applications program and a database. The OO interface receives a message from a client object and performs SQL calls to the database based on the message (see Hembry, col. 6, line 46 to col. 7, line 21).

In contrast, claim 1 recites a combination of steps including receiving a database call, mapping the call to a general computer language programming call of a computer application, and executing the programming call to invoke functions of the computer application that correspond to functions specified by the database call. Hembry cannot teach or suggest these recitations because the client object does not provide a database call to the OO interface. Instead, according to Hembry, a client object provides an object oriented message to the OO interface, which is not a database call

(see Hembry, col. 7, lines 19-21, "client objects communicate with it [OO interface] in the OO manner of passing messages to invoke member functions B"). The only database call taught by Hembry is associated with the communications between the OO interface and the database. Accordingly, the exemplary process taught by Hembry in col. 9, lines 19-27, which is cited by the Examiner, is not the same as the mapping step recited in claim 1 because Hembry mentions mapping an object oriented message to database access mechanisms, and not mapping a database call to a general computer language programming call of a computer application.

Further, according to the Examiner's interpretation of Hembry, there would be no need to perform the mapping and executing steps recited in claim 1. For example, in rejecting claim 1, the Examiner asserts that Hembry teaches an OO applications program that accesses data stored in a relational table by making function calls to the database server of an RDBMS in SQL (see Office Action, page 2, paragraph 3). In such a scenario, however, there would be no need for Hembry to map the database call to a programming language because the database server would only need the database call to access the RDBMS. The C_Obj, depicted in Fig. 2 of Hembry provides the appropriate SQL statements to the database server, and does not itself receive database calls. Therefore, according to the Examiner's interpretation of Hembry, the "computer system" that receives a database call would be the database server, which would simply use the received call to access the database. Accordingly, Hembry cannot teach or suggest the recitations of claim 1 because the reference has no need to perform any mapping functions once the SQL statements are received by the database server.

Additionally, Applicant traverses the Examiner's motivation to modify Hembry to teach the recitations of claim 1. The Examiner asserts that it would have been obvious to one skilled in the art to modify the teachings of Hembry with the steps of mapping the database call to a programming call because it would "improve the accuracy of the database access bridge system and process, and provide a set of computer programs that facilitates the creation management and manipulation of databases" (see Office Action, page 4, lines 1-5). Applicant disagrees. As explained, there is no need for Hembry to map an SQL command once it is received by the database server. Further, it is unclear to Applicant how the implementation of a mapping of a database call would improve the accuracy of the database system taught by Hembry. In contrast, such an implementation may actually decrease the system's accuracy because performing the additional step of mapping an SQL command to a programming language call in Hembry's system would require the database server to translate a received database call from SQL commands to a programming language call, and then back to the same SQL command initially received. This implementation is inefficient and is not needed for Hembry's system to access the database system. For at least the foregoing reasons, one skilled in the art would not have been motivated to modify Hembry to perform the mapping step recited in claim 1.

Because Hembry fails to teach or suggest the recitations of claim 1, Applicant respectfully requests that the rejection of this claim under 35 U.S.C. § 103(a) be withdrawn and the claim allowed.

Claims 3-16 and 18 depend from claim 1. As explained, claim 1 is distinguishable from Hembry. Accordingly, dependent claims 3-16 and 18 are also

distinguishable from <u>Hembry</u> for at least the same reasons set forth for claim 1, and Applicant respectfully requests that the rejection of these claims under 35 U.S.C. § 103(a) be withdrawn and the claims allowed.

Further, <u>Hembry</u> does not teach the computer system is an application server, as recited in claim 4. Contrary to the Examiner's assertions, the database server taught by <u>Hembry</u> is not an application server. <u>Hembry</u> indicates that applications programs accesses data from a database server (see <u>Hembry</u>, col. 4, lines 35-51). Thus, the database server is merely an entity to retrieve data from a database and does not operate as an application server. Also, in further support of the arguments presented above with respect to claim 1, the Examiner is again asserting that the database server is the computer system recited in claims 4 and 1. This interpretation reinforces Applicant's position that <u>Hembry</u> cannot teach or suggest the recitations of claim 1 and 4 because once the database server receives a database call, there would be no need to map the call to another type of call. Accordingly, Applicant respectfully requests that the rejection of claim 4 under 35 U.S.C. § 103(a) be withdrawn and the claim allowed.

Also, <u>Hembry</u> fails to teach or suggest that the database call is receives from a client computer system, as recited in claim 5. As explained, <u>Hembry</u> teaches that an object, such as C_Obj shown in Figs. 3 and 4, sends a message to an OO interface. Only the OO interface provides SQL commands to a database server and the interface is associated with an applications program and not a client system. Accordingly, Applicant respectfully requests that the rejection of claim 5 under 35 U.S.C. § 103(a) be withdrawn and the claim allowed.

Additionally, <u>Hembry</u> does not teach validating data operations prior to issuing an SQL call, as recited in claim 8. Instead, the description presented in col. 3, lines 32-36 of <u>Hembry</u>, which is cited by the Examiner, merely describes function calls that are provided to an RDBMS system as SQL commands. There is nothing in this citation, or anywhere else in <u>Hembry</u>, that refers to validating a data operation prior to using an SQL call, as recited in claim 8. Accordingly, Applicant respectfully requests that the rejection of claim 8 under 35 U.S.C. § 103(a) be withdrawn and the claim allowed.

Also, Applicant asserts that the rejection of claim 12 under the same basis as claim 1 is improper. Claim 12 includes recitations not present in claim 1, thus the Examiner did not address the recitations of this claim. For these foregoing reasons and because <u>Hembry</u> does not teach or suggest determining the methods that are invoked in the objects for use in determining a correspondence between database elements and elements of components that generate database calls, as recited in claim 12, Applicants request that the rejection of this claim be withdrawn and the claim allowed.

Moreover, <u>Hembry</u> fails to teach or suggest identifying the methods by searching for a method of the form <command prefix>XXX, a <get> command prefix, or a <set> command prefix, as recited in claims 13-15. Instead, <u>Hembry</u> merely discloses that the OO interface has knowledge of the database schema of the syntax of SQL calls (see <u>Hembry</u>, col. 7, lines 11-15). <u>Hembry</u> does not mention or suggest identifying methods of any forms, much less the forms recited in claims 13-15. Accordingly, Applicant respectfully requests that the rejection of claims 13-15 under 35 U.S.C. § 103(a) be withdrawn and the claims allowed

The Examiner asserts that <u>Hembry</u> teaches the recitations of claim 18 based on the description of Fig. 2 in col. 6, lines 16-26. Applicant disagrees. This aspect of <u>Hembry</u>'s invention is directed to a two tier system where an applications program sends SQL commands to a database server. The application program itself does not receive any SQL calls. Accordingly, <u>Hembry</u> does not teach or suggest two SQL database calls and a column layout specified in the first call being different that second SQL database call, as recited in claim 18. Accordingly, Applicant respectfully requests that the rejection of claim 18 under 35 U.S.C. § 103(a) be withdrawn and the claim allowed.

Claim 19 includes recitations similar to those included in claim 1. As explained, claim 6 is distinguishable from <u>Hembry</u>. Accordingly, claim 19 is also distinguishable from <u>Hembry</u> for at least the reasons set forth for claim 1, and Applicant respectfully requests that the rejection of this claim under 35 U.S.C. § 103(a) be withdrawn and the claim allowed.

Claims 20-23 depend from claim 19. As explained, claim 19 is distinguishable from <u>Hembry</u>. Accordingly, dependent claims 20-23 are also distinguishable from <u>Hembry</u> for at least the reasons set forth for claim 19, and Applicant respectfully requests that the rejection of these claims under 35 U.S.C. § 103(a) be withdrawn and the claims allowed.

Claims 26 and 32 include recitations similar to those included in claim 1. As explained, claim 1 is distinguishable from <u>Hembry</u>. Accordingly, claims 26 and 32 are also distinguishable from <u>Hembry</u> for at least the same reasons set forth for claim 1. For instance, <u>Hembry</u> does not teach receiving database protocol commands for

accessing a database at an application server and converting the command to a command syntax of the first programming language corresponding to at least a selected one of software components. As explained, Hembry receives a function call at an OO interface, which in turn provides SQL commands to a database server. Thus, the reference cannot teach receiving database protocol commands at an application server and converting the database command to a command syntax, as recited in claim 26. Also, Hembry does not teach a command converter that converts a first database call to a general computer programming language call that is executed to generate a second database call corresponding to the first database call for accessing a database, as recited in claim 32. As explained, Hembry discloses only a single SQL command path, whereby an OO interface provides database commands to a database server. The OO interface does not receive database calls, but instead may receive an object oriented message from an object that requests access to the database.

Because Hembry does not teach or suggest the recitations of claims 26 and 32, Applicant respectfully requests that the rejection of these claims under 35 U.S.C. § 103(a) be withdrawn and the claims allowed.

Claims 27 and 29-31 depend from claim 26. As explained, claim 26 is distinguishable from Hembry. Accordingly, dependent claims 27, and 29-31 are also distinguishable from Hembry for at least the reasons set forth for claim 26, and Applicant respectfully requests that the rejection of these claims under 35 U.S.C. § 103(a) be withdrawn and the claims allowed.

Claims 33-37 and 40 depend from claim 32. As explained, claim 32 is distinguishable from Hembry. Accordingly, dependent claims 33-37 and 40 are also

distinguishable from <u>Hembry</u> for at least the reasons set forth for claim 32, and Applicant respectfully requests that the rejection of these claims under 35 U.S.C. § 103(a) be withdrawn and the claims allowed.

Further, not only does <u>Hembry</u> fails to teach or suggest two database calls, but the reference also fails to teach or suggest two database calls that specify different column names, as recited in claim 37. The discussion in col. 1, lines 27-31 of <u>Hembry</u>, which is cited by the Examiner, merely describes operations associated with conventional database access systems. This discussion is directed to the shortcomings of certain access systems associated with the manipulation of values of methods associated with certain objects. There is no discussion or suggestion in this citation, or anywhere else in <u>Hembry</u>, associated with a first and second database call that each specify different column names, as recited in claim 37. Accordingly, Applicant respectfully requests that the rejection of this claim under 35 U.S.C. § 103(a) be withdrawn and the claim allowed.

Applicants respectfully traverses the rejection of claims 2, 17, 24, 25, 28, 38, and 39 under 35 U.S. C. § 103(a) because <u>Hembry</u> and <u>Moore et al.</u>, alone or in combination, fail to teach or suggest every recitation of these claims.

The Examiner asserts that <u>Hembry</u> teaches all of the recitations of these claims except a programming call being an EJB call and a computer application being a Visual Basic computer application. To compensate for the deficiencies of <u>Hembry</u>, the Examiner asserts that <u>Moore et al.</u> teaches these missing recitations and that one skilled in the art would have been motivated to combine these references to teach or

suggest the recitations of claims 2, 17, 24, 25, 28, 38, and 39. Applicants respectfully disagree.

Although <u>Moore et al.</u> discloses the use of the Java programming language implemented within a Java Runtime environment (see <u>Moore et al.</u>, col. 5, lines 11-53), the reference does not teach EJBs, Visual Basic programming language or either of these program entities being associated with a database call or application. Because neither <u>Hembry</u> or <u>Moore et al.</u> teach or suggest the recitations of claims 2, 17, 24, 25, 28, 38, and 39, the Examiner has failed to establish a prima facie case of obviousness in rejecting these claims. The general disclosure of Java programming in a computer runtime system, as described by <u>Moore et al.</u>, is not only insufficient evidence of EJBs and Visual Basic programming in general, but is also insufficient evidence of a computer language programming call that is mapped to a database call and/or a Visual Basic program that includes the general computer language programming call, as recited in these claims.

Because <u>Hembry</u> and <u>Moore et al.</u>, alone or in combination, fail to teach or suggest every recitation of claims 2, 17, 24, 25, 28, 38, and 39, Applicant respectfully requests that the rejection of these claims under 35 U.S.C. § 103(a) be withdrawn and the claims allowed.

In view of the foregoing remarks, Applicant submits that this claimed invention, is neither anticipated nor rendered obvious in view of the prior art references cited against this application. Applicant therefore request the Examiner's reconsideration and reexamination of the application and the timely allowance of claims 1-40.

Please grant any extensions of time required to enter this response and charge any additional required fees to our deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: December 19, 2002   By: _____

Joseph E. Palys
Reg. No. 46,508